

Proofs for an Abstraction of Continuous Dynamical Systems Utilizing Lyapunov Functions

Christoffer Sloth and Rafael Wisniewski

Abstract—In this report proofs are presented for a method for abstracting continuous dynamical systems by timed automata. The method is based on partitioning the state space of dynamical systems with invariant sets, which form cells representing locations of the timed automata.

To enable verification of the dynamical system based on the abstraction, conditions for obtaining sound, complete, and refinable abstractions are set up.

It is proposed to partition the state space utilizing sub-level sets of Lyapunov functions, since they are positive invariant sets. The existence of sound abstractions for Morse-Smale systems and complete and refinable abstractions for linear systems are proved.

I. INTRODUCTION

Verifying properties such as safety is important for any system. Such verification is based on reachability calculations or approximations. Since the exact reachable sets of continuous and hybrid systems in general are incomputable [1] a lot of attention has been paid to their approximations. Yet reachability is decidable for discrete systems such as automata and timed automata; consequently, there exists a rich set of tools aimed at verifying properties of such systems. Therefore, abstracting dynamical systems by discrete systems would enable verification of dynamical systems using these tools.

There are basically two methods for verifying continuous and hybrid systems. The first is to over-approximate the reachable states by simple convex sets as in [2]. The second method is based on abstracting the original system into a description with reduced complexity, while preserving certain properties of the original systems. This is accomplished for hybrid systems in [3] and for continuous systems in [4].

In this work, continuous systems are abstracted by timed automata. This concept is primarily motivated by [4] where slices are introduced to improve abstractions of continuous systems. A slice is a counterpart of a single direction in continuous systems.

This technical report is devoted to proving the propositions presented in the paper "Abstraction of Continuous Dynamical Systems Utilizing Lyapunov Functions", written by Christoffer Sloth and Rafael Wisniewski for the 49th IEEE Conference on Decision and Control (CDC) [5]. Therefore, that paper can be consulted for further insight in the abstraction method. In that paper the idea of considering both cells

and slices for abstractions was adopted to provide as solution to the following problem.

Problem 1: Given an autonomous dynamical system, find a partition of its state space, which allows arbitrary close over-approximation of the reachable set by a timed automaton.

The abstraction to be addressed preserves safety and has an upper bound on the size of the over-approximation of the reachable set. Furthermore, it is possible to reduce the size of the upper bound to an arbitrary small value, for a class of systems, by refining the partitioning. Hence, we can obtain an abstraction with arbitrary precision of the reachable set.

II. PRELIMINARIES

The purpose of this section is to provide some definitions related to autonomous dynamical systems and timed automata.

An autonomous dynamical system $\Gamma = (X, f)$ is a system with state space $X \subseteq \mathbb{R}^n$ and dynamics described by ordinary differential equations $f : X \rightarrow \mathbb{R}^n$

$$\dot{x} = f(x). \quad (1)$$

The function f is assumed to be locally Lipschitz. Additionally, we assume linear growth of f , then according to Theorem 1.1 in [6] there exists a solution of (1) on $(-\infty, \infty)$.

The solution of (1), from an initial state $x_0 \in X$ at time $t \geq 0$ is described by the flow function $\phi_\Gamma : [0, \epsilon] \times X \rightarrow X$, $\epsilon > 0$ satisfying

$$\frac{d\phi_\Gamma(t, x_0)}{dt} = f(\phi_\Gamma(t, x_0)) \quad (2)$$

for all $t \geq 0$.

Lyapunov functions are utilized in stability theory and are defined in the following [7].

Definition 1 (Lyapunov Function): Assume that a mapping $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is continuous on $G \subset \mathbb{R}^n$ and that G is open and connected. Then a real non-degenerate function $\psi : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{-\infty, \infty\}$ differentiable on G is said to be a Lyapunov function for the differential equation shown in (1) if

p is a critical point of $f \Leftrightarrow p$ is a critical point of ψ

$$\dot{\psi}(x) = \sum_{j=1}^n \frac{\partial \psi}{\partial x_j}(x) f^j(x) \quad (3)$$

$$\psi(x) : \begin{cases} = 0 & \text{if } x = p \\ < 0 & \text{if } x \in G \setminus \{p\} \end{cases} \quad (4)$$

This work was supported by MT-LAB, a VKR Centre of Excellence. Christoffer Sloth is with Department of Computer Science, Aalborg University, 9220 Aalborg East, Denmark csloth@cs.aau.dk. Rafael Wisniewski is with the Section of Automation & Control, Aalborg University, 9220 Aalborg East, Denmark raf@es.aau.dk.

and $\exists \alpha > 0$ and an open neighborhood of each critical point p , where

$$||\dot{\psi}(x)|| \geq \alpha ||x - p||. \quad (5)$$

Note that we do not require positive definiteness of ψ .

Definition 2 (Reachability for Dynamical System): The reachable set of a dynamical system Γ from a set of initial states $X_0 \subseteq X$ on the time interval $[t_1, t_2]$ is defined as

$$\text{Reach}_{[t_1, t_2]}(\Gamma, X_0) = \{x \in X | \exists t \in [t_1, t_2], \exists x_0 \in X_0, \text{ such that } x = \phi_\Gamma(t, x_0)\}. \quad (6)$$

The dynamical system will be abstracted by a timed automaton. Therefore, a definition of timed automaton is provided in the following [8]. In the definition, a set of clock constraints $\Psi(C)$ for the set C of clocks is utilized. $\Psi(C)$ contains all invariants and guards of the timed automaton, consequently it is described by the following grammar [9]:

$$\begin{aligned} \psi &::= c_1 \bowtie k | c_1 - c_2 \bowtie k | \psi_1 \wedge \psi_2, \text{ where} \\ c_1, c_2 &\in C, k \in \mathbb{Z}, \text{ and } \bowtie \in \{\leq, <, =, >, \geq\}. \end{aligned} \quad (7)$$

Note that the clock constraint k should be an integer, but in this paper no effort is done in converting the clock constraints into integers.

Definition 3 (Timed Automaton): A timed automaton, \mathcal{A} , is a tuple $(L, L_0, C, \Sigma, I, \Delta)$, where

- L is a finite set of locations, and $L_0 \subseteq L$ is the set of initial locations.
- C is a finite set of clocks all with values in $\mathbb{R}_{\geq 0}$.
- Σ is the input alphabet.
- $I : L \rightarrow \Psi(C)$ assigns invariants to locations, where $\Psi(C)$ is the set of all clock constraints, see (7).
- $\Delta \subseteq L \times \Psi(C) \times \Sigma \times 2^C \times L$ is a finite set of transition relations. The transition relations provide edges between locations as tuples $(l, G_{l \rightarrow l'}, \sigma, R_{l \rightarrow l'}, l')$, where l is the source location, l' is the destination location, $G_{l \rightarrow l'} \in \Psi(C)$ is the guard set, σ is a symbol in the alphabet Σ , and $R_{l \rightarrow l'} \in 2^C$ gives the set of clocks to be reset.

We use the mapping $v : C \rightarrow \mathbb{R}_{\geq 0}$ for a clock valuation on a set of clocks C . Additionally, the initial valuation is denoted v_0 , where $v_0(c) = 0$ for all $c \in C$.

Analog to the solution of (1) shown in (2), a run of a timed automaton is defined in the following.

Definition 4 (Run of Timed Automaton): A run of a timed automaton \mathcal{A} is a possibly infinite sequence of alternations between time steps and discrete steps in the following form

$$(v_0, l_0) \xrightarrow{t_1} (v_0 + t_1, l_0) \xrightarrow{\sigma_1} (v_1, l_1) \longrightarrow \dots \quad (8)$$

The multifunction describing a run of a timed automaton is $\phi_{\mathcal{A}} : \mathbb{R}_{\geq 0} \times L_0 \rightarrow 2^L$. Here $l \in \phi_{\mathcal{A}}(t, l_0)$ if and only if the timed automaton \mathcal{A} initialized in l_0 can be in location l at time $t = \sum_i t_i$.

From the run of a timed automaton, the reachable set is defined below.

Definition 5 (Reachability for Timed Automaton): The reachable set of a timed automaton \mathcal{A} with initial locations

L_0 on the time interval $[t_1, t_2]$ is defined as

$$\text{Reach}_{[t_1, t_2]}(\mathcal{A}, L_0) = \{l \in L | \exists t \in [t_1, t_2], \exists l_0 \in L_0, \text{ such that } l \in \phi_{\mathcal{A}}(t, l_0)\}. \quad (9)$$

III. GENERATION OF FINITE PARTITION

A finite partition of the state space of the considered system is generated using slices, which are set-differences between positive invariant sets.

Proposition 1: If $S_1 \cap S_2 \neq \emptyset$ then

$$\text{int}(S_1 \cap S_2) \neq \emptyset. \quad (10)$$

Proof: Let $p \in \text{bd}(S_1) \cap \text{bd}(S_2)$ by Theorem 7.7 in [10] there exists a local coordinate system (Y, U) such that

$$Y(S_1 \cap U) = H_1^+ \subset \mathbb{R}^n \quad (11a)$$

$$Y(S_2 \cap U) = H_2^+ \subset \mathbb{R}^n \quad (11b)$$

where H_1^+ and H_2^+ are supporting hyperplanes of S_1 and S_2 . Thus $\dim(H_1^+ \cap H_2^+) = n$. ■

Note that the intersection of slices may form multiple disjoint sets. Therefore, the intersection of k slices is denoted an extended cell $e_{\text{ex},g}$. Each of the disjoint sets of an extended cell $e_{\text{ex},g}$ is called a cell $e_{g,h}$.

IV. GENERATION OF TIMED AUTOMATON FROM FINITE PARTITION

A timed automaton is generated by associating each cell of a partition with a location and by inserting guards and invariants calculated based on the dynamics. The method is presented in [5] and is very similar to the method presented in [4].

Proposition 2: $\mathcal{A}(\mathcal{S})$ is a deterministic timed automaton, if and only if for each cell $e_{(g,h)}$ and for all $i = 1, \dots, k$ the set

$$e_{(g,h)} \cap \psi_i^{-1}(a_{(i,g_i-1)}) \quad (12)$$

is connected.

Proof: If $e_{(g,h)} \cap \psi_i^{-1}(a_{(i,g_i-1)})$ is not connected for some i , then σ_i is the label of multiple outgoing transitions from the location $e_{(g,h)}$, i.e. there exist multiple transitions in Δ , where $e_{(g,h)}$ is the source location and σ_i is the label. Therefore, the timed automaton $\mathcal{A}(\mathcal{S})$ is nondeterministic. ■

Proposition 3: Let $\mathcal{A}_{\text{ex}}(\mathcal{S})$ be a timed automaton, with locations associated to extended cells, and let the slices of \mathcal{S} be generated such that for each pair $S_{(i,g_i)}$ and $S_{(j,g_j)}$, with $i, j \in \{1, \dots, k\}$, $g_i \in \{1, \dots, |\mathcal{S}_i|\}$, $g_j \in \{1, \dots, |\mathcal{S}_j|\}$, we have

$$S_{(i,g_i)} \cap S_{(j,g_j)} \neq \emptyset \quad \forall i \neq j. \quad (13)$$

Then $\mathcal{A}_{\text{ex}}(\mathcal{S})$ is isomorphic to the parallel composition of k timed automata each generated by one slice-family \mathcal{S}_i .

Proof: Consider the timed automaton $\mathcal{A}_{||}(\mathcal{S}) = \mathcal{A}_1(\mathcal{S}_1) || \dots || \mathcal{A}_k(\mathcal{S}_k)$ where $\mathcal{A}_i(\mathcal{S}_i) = (L_i, L_{0,i}, C_i, \Sigma_i, I_i, \Delta_i)$ and $L_i = \{l_{(i,1)}, \dots, l_{(i,|\mathcal{S}_i|)}\}$, abstracting the slices $S_{(i,1)}, \dots, S_{(i,|\mathcal{S}_i|)}$. Then the timed automaton $\mathcal{A}_{||}(\mathcal{S})$ is given by

- **Locations:** $L = L_1 \times \dots \times L_k$, which according to Definition 10 in [5] represents extended cells, if the transversal intersection of all slices is nonempty i.e. (13) is satisfied.
- **Clocks:** $C = \{c_i, \dots, c_k\}$, where c_i monitors the time for being in a slice of S_i .
- **Invariants:** The invariant for location $l_{\text{ex},g} = (l_{(1,g_1)}, \dots, l_{(k,g_k)})$ is identical to (18) in [5] and is

$$I(l_{\text{ex},g}) = \bigwedge_{i=1}^k I_i(l_{(i,g_i)}). \quad (14)$$

- **Input Alphabet:** $\Sigma = \{\sigma_1, \dots, \sigma_k\}$.
- **Transition relations:** Σ_i is disjoint from Σ_j for all $i \neq j$; hence, item 1) in Definition 15 in [5] never happens.

This implies that $\mathcal{A}_{\parallel}(\mathcal{S}) = \mathcal{A}_1(\mathcal{S}_1) \parallel \dots \parallel \mathcal{A}_k(\mathcal{S}_k)$ and $\mathcal{A}_{\text{ex}}(\mathcal{S})$ are isomorph. ■

Proposition 5: Let $\mathcal{S} = \{S_1, \dots, S_k\}$ be a collection of slice-families, and ψ_i be a partitioning function for S_i . The timed automata $\mathcal{A}_{\text{ex}}(\mathcal{S})$ and $\mathcal{A}(\mathcal{S})$ are bisimilar if for each cell $e_{(g,h)} \in K(\mathcal{S})$ and each $i \in \{1, \dots, k\}$

$$e_{(g,h)} \cap \psi_i^{-1}(a_{(i,g_i-1)}) \neq \emptyset \quad \forall h \text{ or} \quad (15a)$$

$$e_{(g,h)} \cap \psi_i^{-1}(a_{(i,g_i-1)}) = \emptyset \quad \forall h. \quad (15b)$$

If (15) holds, then all cells in each extended cell have the same symbols on their outgoing transitions.

Proof: Let $e_{(g,h)}$ with $h = 1, \dots, m$ be the cells which union is the extended cell $e_{\text{ex},g}$. Then

$$I(e_{(g,h)}) = I(e_{(g,k)}) \quad \forall h, k \in \{1, \dots, m\} \quad (16)$$

as the invariants are calculated based on slices (18) in [5].

If the partition satisfies (15), then the same outgoing transitions exist for all cells within the same extended cell. Furthermore,

$$G_{(g,h) \rightarrow (g',h')} = G_{(g,k) \rightarrow (g',k')} \quad \forall h, k \in \{1, \dots, m\} \quad (17)$$

since the guards are also calculated based on slices (19b) in [5]. This implies that all possible behaviors from each cell in an extended cell are the same; hence, $\mathcal{A}(\mathcal{S})$ is bisimilar to a timed automaton $\mathcal{A}_{\text{ex}}(\mathcal{S})$. ■

V. CONDITIONS FOR THE PARTITIONING

A sound and a complete abstraction of a dynamical system is illustrated in Fig. 1. Definitions of sound and complete

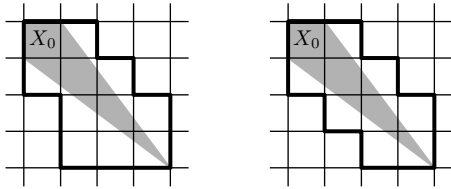


Fig. 1. Illustration of the reachable set of a dynamical system (gray) from initial set X_0 and a sound approximation of this (cells within bold black lines) on the left and a complete abstraction on the right.

abstractions are available in [5].

Proposition 6: A timed automaton $\mathcal{A}_{\text{ex}} = \mathcal{A}_1 \parallel \dots \parallel \mathcal{A}_k$, with locations abstracting extended cells, is a sound (complete) abstraction of the system Γ if and only if $\mathcal{A}_1, \dots, \mathcal{A}_k$ are sound (complete) abstractions of Γ .

Proof: If the locations of \mathcal{A}_{ex} are extended cells, then soundness of \mathcal{A}_{ex} can be reformulated to the following.

A timed automaton \mathcal{A}_{ex} with $L_0 = \{e_{\text{ex},g} | g \in \mathcal{G}_0 \subseteq \mathcal{G}\}$ is said to be a sound abstraction of Γ with $X_0 = \bigcup_{g \in \mathcal{G}_0} e_{\text{ex},g}$ on $[t_1, t_2]$ if for all $t \in [t_1, t_2]$ and for all $g \in \mathcal{G}$

$$\bigcap_{i=1}^k S_{(i,g_i)} \cap \text{Reach}_{[t,t]}(\Gamma, X_0) \neq \emptyset \quad \text{implies} \quad (18a)$$

$\exists l_0 \in L_0$ such that

$$\bigcap_{i=1}^k S_{(i,g_i)} \in \alpha_K^{-1}(\phi_{\mathcal{A}_{\text{ex}}}(t, l_0)) \quad (18b)$$

which is equivalent to: For all $i = \{1, \dots, k\}$, all $g \in \mathcal{G}$, and for all $t \in [t_1, t_2]$

$$S_{(i,g_i)} \cap \text{Reach}_{[t,t]}(\Gamma, X_0) \neq \emptyset \quad \text{implies} \quad (19a)$$

$\exists l_{0,i} \in L_{0,i}$ such that

$$\alpha_K^{-1}(\phi_{\mathcal{A}_i}(t, l_{0,i})) = S_{(i,g_i)}. \quad (19b)$$

From (19) it is seen that $\mathcal{A}_{\text{ex}} = \mathcal{A}_1 \parallel \dots \parallel \mathcal{A}_k$ is sound if and only if \mathcal{A}_i is sound for $i = 1, \dots, k$. Similar arguments can be used to prove the completeness part of Proposition 6. ■

Proposition 7 (Sufficient Condition for Soundness): A timed automaton $\mathcal{A}(\mathcal{S})$ is a sound abstraction of the system Γ , if its invariants and guards are formed using

$$\underline{t}_{S_{(i,g_i)}} \leq \frac{|a_{(i,g_i)} - a_{(i,g_i-1)}|}{\sup\{|\psi_i(x)| \in \mathbb{R}_{\geq 0} | x \in S_{(i,g_i)}\}} \quad (20a)$$

$$\bar{t}_{S_{(i,g_i)}} \geq \frac{|a_{(i,g_i)} - a_{(i,g_i-1)}|}{\inf\{|\psi_i(x)| \in \mathbb{R}_{\geq 0} | x \in S_{(i,g_i)}\}} \quad (20b)$$

where $\psi_i(x)$ is defined as shown in (3).

Proof: Let $\mathcal{A}(\mathcal{S})$ be a timed automaton with $L_0 = \{e_i | i \in \mathcal{I}\}$, be an abstraction of Γ with initial set $X_0 = \bigcup_{i \in \mathcal{I}} e_i$. If guards and invariants of $\mathcal{A}(\mathcal{S})$ satisfy (20), then

$$\text{Reach}_{[t_1, t_2]}(\Gamma, X_0) \subseteq \alpha_K^{-1}(\text{Reach}_{[t_1, t_2]}(\mathcal{A}, L_0)) \quad (21)$$

since for all $x_0 \in \psi_i^{-1}(a_{(i,g_i)})$ there exists $t \in [\underline{t}_{S_{(i,g_i)}}, \bar{t}_{S_{(i,g_i)}}]$ such that

$$\phi_{\Gamma}(t, x_0) \in \psi_i^{-1}(a_{(i,g_i-1)}). \quad (22)$$

■

Proposition 8 (Sufficient Condition for Completeness): Let $\mathcal{S} = \{S_i | i = 1, \dots, k\}$ be a collection of slice-families and let

$$S_{(i,g_i)} = \psi_i^{-1}([a_{(i,g_i-1)}, a_{(i,g_i)}]). \quad (23)$$

A deterministic timed automaton is a complete abstraction if

- 1) $\bar{t}_{S_{(i,g_i)}} = \underline{t}_{S_{(i,g_i)}} = t_{(i,g_i)}$ and
- 2) for any $g \in \mathcal{G}$ with $g_i \geq 2$ there exists a time $t_{(i,g_i)}$ such that $\forall x_0 \in \psi_i^{-1}(a_{(i,g_i)})$

$$\phi_{\Gamma}(t_{(i,g_i)}, x_0) \in \psi_i^{-1}(a_{(i,g_i-1)}). \quad (24)$$

Proof: The proposition states that it takes the same time for all trajectories of Γ to propagate between any two level sets of ψ_i . From this it follows that $\mathcal{A}(\mathcal{S})$ is complete if $\bar{t}_{S(i,g_i)}$ and $\underline{t}_{S(i,g_i)}$ are equal to $t_{(i,g_i)}$. ■

Proposition 9 (Nec. Cond. for Refinable Abstraction): If $\mathcal{A}(\mathcal{S})$ is a refinable abstraction of a system Γ , then \mathcal{S} is a collection of n slice-families.

Proof: If $\mathcal{A}(\mathcal{S})$ is a refinable abstraction, then for any $\epsilon > 0$ there exists a partitioning $K(\mathcal{S})$ such that (30) in [5] holds for cells in $K(\mathcal{S})$. Therefore,

$$S_{(i,g_i)} \subset \psi_i^{-1}(a_{(i,g_i)}) + B(\epsilon) \quad (25)$$

where $\epsilon > 0$. Note that $a_{(i,g_i)}$ is a regular value of ψ_i , i.e. the dimension of the level set $\psi_i^{-1}(a_{(i,g_i)})$ is $n - 1$. The locations of $\mathcal{A}(\mathcal{S})$ are cells for which

$$\bigcup_h e_{(g,h)} = \bigcap_{i=1}^k S_{(i,g_i)} \quad (26a)$$

$$\subset \bigcap_{i=1}^k (\psi_i^{-1}(a_{(i,g_i)}) + B(\epsilon)) \quad (26b)$$

$$\subset \bigcap_{i=1}^k \psi_i^{-1}(a_{(i,g_i)}) + B(2\epsilon). \quad (26c)$$

But (26c) is true for any ϵ , thus it is enough to prove that

$$\dim(\bigcap_{i=1}^k \psi_i^{-1}(a_{(i,g_i)})) = 0. \quad (27)$$

Using Theorem 7.7 in [10] the dimension of an extended cell is given by

$$\begin{aligned} \dim(\bigcap_{i=1}^k \psi_i^{-1}(a_{(i,g_i)})) \\ = [(n-1) + (n-1) - n] + (n-1) - n \\ + (n-1) - n \dots \end{aligned} \quad (28a)$$

$$= k(n-1) - (k-1)n. \quad (28b)$$

We see that if $k \neq n$ then $\dim(\bigcap_{i=1}^k \psi_i^{-1}(a_{(i,g_i)})) \neq 0$, thus we have contradiction. We conclude that $k = n$. ■

VI. PARTITIONING THE STATE SPACE USING LYAPUNOV FUNCTIONS

Positive invariant sets are used in stability theory in the form of sub-level sets of Lyapunov functions. This concept is adopted in this work to synthesize partitions.

Definition 6: Two Lyapunov functions $\psi_1, \psi_2 : \mathbb{R}^n \rightarrow \mathbb{R}$ are transversal if the level sets $\psi_1^{-1}(a)$ and $\psi_2^{-1}(a)$ are transversal for any $a \in \mathbb{R} \setminus \{0\}$.

Proposition 10: Let $n > 1$. For any Morse-Smale system (see Chapter 4 in [11]) on \mathbb{R}^n there exists n transversal Lyapunov functions.

Proof: Let $S(n, \mathbb{R})$ be a set of $n \times n$ symmetric matrices. $S(n, \mathbb{R})$ is a subspace of $M(n, \mathbb{R})$ of $\dim(S(n, \mathbb{R})) = n(n+1)/2$. Consider the map $\psi_A : S(n, \mathbb{R}) \rightarrow S(n, \mathbb{R})$ and let

$$P \mapsto A^T P + P A. \quad (29)$$

Now consider the map $\det : M(n, \mathbb{R}) \rightarrow \mathbb{R}$ and let

$$A \mapsto \det(A). \quad (30)$$

Then $(\det \circ \psi_A)^{-1}(\{0\})$ is a closed set. Therefore,

$$U_A \equiv \{P \in S(n, \mathbb{R}) | \det \circ \psi_A(P) \neq 0\} \quad (31)$$

is an open set. $V_A \equiv V \cap U_A$ is open, where

$$V = \{P \in S(n, \mathbb{R}) | \det(P) \neq 0\} \quad (V \text{ is open}). \quad (32)$$

Let $\Theta = \{Q \in S(n, \mathbb{R}) | Q > 0\}$ by Proposition 2.18 in [11] the map

$$M(n, \mathbb{R}) \rightarrow C^n/S^n \text{ defined by} \quad (33)$$

$$L \mapsto \text{diag}([\lambda_1, \dots, \lambda_n]) \text{ is continuous.} \quad (34)$$

Thus Θ is an open set in $S(n, \mathbb{R})$.

We pick an open neighborhood around $Q = A^T P + P A$ and denote it U . Then for every $Q' \in U$ there exists a (unique) P , thus $\psi_A^{-1}(U)$ is a nonempty open set in $S(n, \mathbb{R})$.

We can pick n linear independent matrices $P_1, \dots, P_n \in \psi_A^{-1}(U)$. This is possible because $\psi_A^{-1}(U)$ is open in $S(n, \mathbb{R})$ and $\dim(S(n, \mathbb{R}))$ is $n(n+1)/2$. Then for any $a \in \mathbb{R} \setminus \{0\}$ and $i \neq j$

$$\{x \in \mathbb{R}^n | x^T P_i x = a\} \cap \{x \in \mathbb{R}^n | x^T P_j x = a\}. \quad (35)$$

Extending this to Morse-Smale systems follows directly from Theorem 1 in [7]. ■

A. Complete Abstraction

A complete abstraction of (1) can be obtained by constructing a partition generated by Lyapunov functions, which satisfies Proposition 8.

Proposition 11: Let each slice-family of $\mathcal{S} = \{\mathcal{S}_i | i = 1, \dots, k\}$ be associated with a Lyapunov function $\psi_i(x)$ for the system Γ , such that $S_{(i,j)} = \psi_i^{-1}([a_{(i,j-1)}, a_{(i,j)}])$ and let

$$\psi_i(x) = \alpha \dot{\psi}_i(x) \quad \forall x \in \mathbb{R}^n. \quad (36)$$

Then $\mathcal{A}(\mathcal{S})$ is a complete abstraction of Γ .

Proof: Let $\psi(x)$ be a Lyapunov function for the system Γ and let $x, x' \in \psi^{-1}(a_m)$. According to Proposition 8 the abstraction is complete if there exists a t_m , for $m = 2, \dots, k$ such that

$$\phi_\Gamma(t_m, x), \phi_\Gamma(t_m, x') \in \psi^{-1}(a_{m-1}). \quad (37)$$

This is true if

$$\dot{\psi}(\phi_\Gamma(t, x)) - \dot{\psi}(\phi_\Gamma(t, x')) = 0 \quad \forall t. \quad (38)$$

The combination of (37) and (38) implies that for all $c > 0$ there exists an α such that

$$\psi^{-1}(c) = \dot{\psi}^{-1}\left(\frac{c}{\alpha}\right) \quad (39)$$

hence for all x there exists an α such that

$$\psi(x) = \alpha \dot{\psi}(x). \quad (40)$$

Proposition 12: For any hyperbolic linear system Γ there exists n transversal Lyapunov functions $\psi_i(x)$ each satisfying

$$\psi_i(x) = \alpha \dot{\psi}_i(x) \quad \forall x \in \mathbb{R}^n. \quad (41)$$

Proof: This is proved for linear systems, by constructing the complete abstraction.

Consider a linear differential equation

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \lambda_1 I_1 & 0 \\ 0 & \lambda_2 I_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (42)$$

where I_1, I_2 are identity matrices and $\lambda_1 < 0$ and $\lambda_2 > 0$.

The stable and unstable subspaces of (42) are orthogonal and can be treated separately. This system is divided into a stable space described by x_1 and an unstable space described by x_2 . For $i \in \{1, 2\}$ let $\psi_i(x_i) = x_i^T P_i x_i$ be a quadratic Lyapunov function. Then its derivative is $\dot{\psi}(x_i) = x_i^T Q_i x_i$, where

$$2\lambda_i P_i = Q_i \quad \text{for } i = 1, 2. \quad (43)$$

This implies that any quadratic Lyapunov function satisfies Proposition 11 and hence generates a complete abstraction.

Since hyperbolic linear systems are topologically conjugate if and only if they have the same index [12]. There is a homeomorphism $h : \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that any hyperbolic linear system is topologically conjugate of (42), by choosing I_1 and I_2 appropriately. Note that h is a diffeomorphism on $\mathbb{R}^n \setminus \{0\}$.

This implies that there exists a complete abstraction of every hyperbolic linear system. ■

VII. CONCLUSION

In this report proofs associated with a method for abstracting hyperbolic dynamical systems by timed automata have been presented. The method is based on partitioning the state space of the dynamical systems by set-differences of invariant sets.

To enable both verification and falsification of safety properties for the considered system based on the abstraction, conditions for soundness, completeness, and refinability have been set up. Furthermore, it is shown that the abstraction can be obtained as a parallel composition of multiple timed automata under certain conditions.

Finally, it is shown that there exist sound and refinable abstractions for hyperbolic Morse-Smale systems. Additionally, it is shown that there exist complete and refinable abstractions for any hyperbolic linear systems.

REFERENCES

- [1] E. Asarin, T. Dang, G. Frehse, A. Girard, C. L. Guernic, and O. Maler, "Recent progress in continuous and hybrid reachability analysis," in *Proceedings of the 2006 IEEE Conference on Computer Aided Control Systems Design*, 2006, pp. 1582–1587.
- [2] A. B. Kurzhanski and I. Vlyi, *Ellipsoidal Calculus for Estimation and Control*. Birkhäuser Boston, 1997.
- [3] A. Tiwari, "Abstractions for hybrid systems," *Formal Methods in System Design*, pp. 57–83, 2008.
- [4] O. Maler and G. Batt, "Approximating continuous systems by timed automata," in *Proceedings of the 1st international workshop on Formal Methods in Systems Biology*, 2008, pp. 77–89.
- [5] C. Sloth and R. Wisniewski, "Abstraction of continuous dynamical systems utilizing lyapunov functions," in *Proceedings of the 49th IEEE Conference on Decision and Control*, Atlanta, Georgia, USA, December 2010.
- [6] F. Clarke, Y. Ledyev, R. Stern, and P. Wolenski, *Nonsmooth Analysis and Control Theory*, A. Axler, F. Gehring, and K. Ribet, Eds. Springer, 1998.
- [7] K. R. Meyer, "Energy functions for morse-smale systems," *American Journal of Mathematics*, vol. 90, no. 4, pp. 1031–1040, 1968.
- [8] R. Alur and D. L. Dill, "A theory of timed automata," *Theoretical Computer Science*, vol. 126, no. 2, pp. 183–235, April 1994.
- [9] C. Thrane, U. Fahrenberg, and K. G. Larsen, "Quantitative analysis of weighted transition systems," *Journal of Logic and Algebraic Programming*, 2010.
- [10] G. E. Bredon, *Topology and Geometry*. Springer, 1993.
- [11] J. P. Junior and W. de Melo, *Geometric Theory of Dynamical Systems: An Introduction*. Springer, 1980.
- [12] M. W. Hirsch, S. Smale, and R. L. Devaney, *Differential Equations, Dynamical Systems & An Introduction to Chaos*, 2nd ed. Elsevier, 2004.